

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

INVENTOR(S): Naoyuki KAMEI, et al.

U.S.S.N.: Not Yet Assigned

ART UNIT: Not Yet Assigned

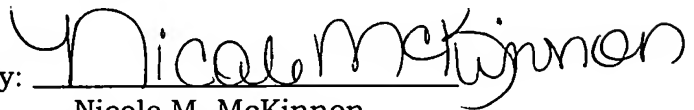
FILED: HERewith

EXAMINER: Not Yet Assigned

FOR: CPU, INFORMATION PROCESSING DEVICE INCLUDING THE CPU, AND
CONTROLLING METHOD OF CPU

CERTIFICATE OF EXPRESS MAILING (Label No.: EV 342614011 US)

I certify that this document fee is being deposited with the U.S. Postal Service
"Express Mail Post Office to Addressee" service under 37 C.F.R. in an envelope
addressed to Mail Stop PATENT APPLICATION, Commissioner for Patents, P.O. Box
1450, Alexandria, VA 22313-1450 on November 21, 2003.

By: 
Nicole M. McKinnon

Mail Stop PATENT APPLICATION
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

TRANSMITTAL OF CERTIFIED COPIES

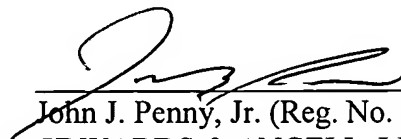
Sir:

Attached please find two certified copies of the foreign application from which priority is
claimed for this case:

Country: JAPAN
Application Number: 2002-340055
Filing Date: 22 November 2002

Respectfully submitted,

Date: November 21, 2003
Customer No. 21874


John J. Penny, Jr. (Reg. No. 36,984)
EDWARDS & ANGELL, LLP
P.O. Box 9169
Boston, MA 02209
Tel: (617) 439-4444

日本国特許庁
JAPAN PATENT OFFICE

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office.

出願年月日
Date of Application: 2002年11月22日

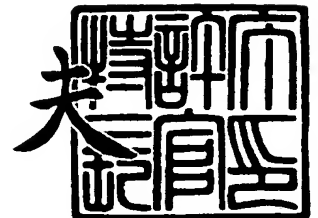
出願番号
Application Number: 特願2002-340055
[ST. 10/C]: [JP2002-340055]

出願人
Applicant(s): シャープ株式会社

2003年10月14日

特許庁長官
Commissioner,
Japan Patent Office

今井康夫



出証番号 出証特2003-3084393

【書類名】 特許願

【整理番号】 02J03452

【提出日】 平成14年11月22日

【あて先】 特許庁長官 殿

【国際特許分類】 G06F 1/00
G06F 9/06
G06F 9/45

【発明者】

【住所又は居所】 大阪府大阪市阿倍野区長池町 2 2 番 2 2 号 シャープ株式会社内

【氏名】 亀井 直幸

【発明者】

【住所又は居所】 大阪府大阪市阿倍野区長池町 2 2 番 2 2 号 シャープ株式会社内

【氏名】 ▲吉▼村 創一

【発明者】

【住所又は居所】 大阪府大阪市阿倍野区長池町 2 2 番 2 2 号 シャープ株式会社内

【氏名】 西村 道明

【発明者】

【住所又は居所】 大阪府大阪市阿倍野区長池町 2 2 番 2 2 号 シャープ株式会社内

【氏名】 水山 善雄

【発明者】

【住所又は居所】 大阪府大阪市阿倍野区長池町 2 2 番 2 2 号 シャープ株式会社内

【氏名】 木下 浩樹

【発明者】

【住所又は居所】 大阪府大阪市阿倍野区长池町 2 2 番 2 2 号 シャープ株式会社内

【氏名】 南 崇博

【特許出願人】

【識別番号】 000005049

【氏名又は名称】 シャープ株式会社

【代理人】

【識別番号】 100080034

【弁理士】

【氏名又は名称】 原 謙三

【電話番号】 06-6351-4384

【選任した代理人】

【識別番号】 100113701

【弁理士】

【氏名又は名称】 木島 隆一

【選任した代理人】

【識別番号】 100115026

【弁理士】

【氏名又は名称】 圓谷 徹

【選任した代理人】

【識別番号】 100116241

【弁理士】

【氏名又は名称】 金子 一郎

【手数料の表示】

【予納台帳番号】 003229

【納付金額】 21,000円

【提出物件の目録】

【物件名】 明細書 1

【物件名】 図面 1

【物件名】 要約書 1

【包括委任状番号】 0208489

【プルーフの要否】 要

【書類名】 明細書

【発明の名称】 CPUおよびこれを備えた情報処理装置、CPUの制御方法

【特許請求の範囲】

【請求項 1】

内部にキャッシュを備えており、上記キャッシュに対してデータの書き換えを行うとともに、この書き換えたデータを所望のタイミングで外部メモリに反映させるライトバック制御を行うCPUにおいて、

処理するタスクの必要メモリ量に応じて、上記キャッシュだけにアクセスしてタスクの処理が可能であるか否かを判定し、該処理が可能である場合には、上記外部メモリに対するアクセスを禁止する制御手段を備えていることを特徴とするCPU。

【請求項 2】

上記制御手段は、上記キャッシュの残り容量および／または処理するタスクの必要メモリ量を検出することを特徴とする請求項 2 に記載のCPU。

【請求項 3】

上記外部メモリに対するアクセスが禁止されている状態において、上記制御手段が、上記キャッシュだけでは処理が不可能と判断した場合、あるいはキャッシュミスが発生した場合には、上記外部メモリに対するアクセス禁止を解除することを特徴とする請求項 2 に記載のCPU。

【請求項 4】

内部クロックのクロック周波数を制御するクロック制御手段を備えており、
該クロック制御手段は、上記外部メモリに対するアクセスが禁止された場合には、上記クロック周波数を変化させることを特徴とする請求項 1 ～ 3 の何れか 1 項に記載のCPU。

【請求項 5】

上記制御手段は、上記キャッシュ内における不要なデータを格納しているアドレスを検出し、該検出されたアドレスに対応するキャッシュ領域を解放することを特徴とする請求項 1 ～ 4 の何れか 1 項に記載のCPU。

【請求項 6】

電源投入後の初期動作時においては、所望のプログラムおよびデータを上記外部メモリから上記キャッシュに読み込んだ上で、上記外部メモリへのアクセスを禁止することを特徴とする請求項 1 ～ 5 の何れか 1 項に記載の C P U。

【請求項 7】

上記制御手段は、タスクの状況が変化する際に、上記外部メモリに対するアクセスが必要か否かについての判断を行うことを特徴とする請求項 1 ～ 6 の何れか 1 項に記載の C P U。

【請求項 8】

上記制御手段は、上記キャッシュ内のプログラムおよびデータについて、パージされたか否かについて判定し、パージされていない場合には、上記プログラムおよびデータを上記外部メモリから読み込む処理を行わないことを特徴とする請求項 1 ～ 7 の何れか 1 項に記載の C P U。

【請求項 9】

請求項 1 ～ 8 に記載の C P U と、外部メモリと、該外部メモリへ電力を供給する電源供給手段とを備えた情報処理装置において、

上記電源供給手段は、上記外部メモリへのアクセスが禁止されている場合には、該外部メモリへの電力供給を停止することを特徴とする情報処理装置。

【請求項 1 0】

上記外部メモリは、複数のモジュールを含んでおり、

上記制御手段は、上記モジュールごとに電力供給の制御を行うことを特徴とする請求項 9 に記載の情報処理装置。

【請求項 1 1】

内部に備えたキャッシュに対してデータの書き換えを行うとともに、この書き換えたデータを所望のタイミングで外部メモリに反映させるライトバック制御を行う C P U の制御方法において、

処理するタスクの必要メモリ量に応じて、上記キャッシュだけにアクセスしてタスクの処理が可能であるか否かを判定し、該処理が可能であると判定した場合には、上記外部メモリに対するアクセスを禁止することを特徴とする C P U の制御方法。

【発明の詳細な説明】**【0001】****【発明の属する技術分野】**

本発明は、CPU (Central Processing Unit) に対する電力供給を停止することなく、外部メモリに対する供給電力を削減して、CPU全体の消費電力を低減することが可能なCPUおよびこれを備えた情報処理装置、CPUの制御方法に関するものである。

【0002】**【従来の技術】**

従来、例えば、PC (Personal Computer) のような情報処理装置において、CPUの外部に備えられる外部メモリなどを停止して省電力化を図ることが提案されている。

【0003】

ここで、PCが備えているCPUの内部には、キャッシュと呼ばれる高速小容量のメモリが備えられている。一方、CPUの外部には、DRAMのような低速大容量の外部メモリが設けられている。

【0004】

そして、CPUは、キャッシュと外部メモリとを適宜使い分けて各処理を行っており、例えば、通常はキャッシュだけを書き換え、必要に応じてキャッシュの書き換えられた内容を外部メモリに反映させる、いわゆるライトバック制御を行う。

【0005】

このようなライトバック制御を行うCPUを備えた情報処理装置の一例としては、レジューム時に必要なプログラムを予めCPUのキャッシュにロードしておき、復帰時に外部メモリが使用可能になったことのチェックを、外部のハードウェアではなくキャッシュに書き込まれたプログラムを用いて処理を行うことで、消費電力を抑えることが可能な情報処理装置が提案されている（特許文献1参照）。

【0006】

同様に、ダイナミック素子の内部状態をスタティック素子に退避して電源を切断し、次の電源投入時に退避した内部状態を復元してレジューム機能を実現する情報処理装置が提案されている（特許文献 2 参照）。

【 0 0 0 7 】**【特許文献 1】**

特開平 1 1 - 2 7 2 3 4 7 号公報（公開日：1 9 9 9 年 1 0 月 8 日）

【 0 0 0 8 】**【特許文献 2】**

特開平 6 - 2 3 0 8 4 5 号公報（公開日：1 9 9 4 年 8 月 1 9 日）

【 0 0 0 9 】**【発明が解決しようとする課題】**

しかしながら、上記従来の情報処理装置は、消費電力を削減するためのレジューム時に外部メモリ等に対する電力供給を停止しているものの、CPU が処理を行う場合には、外部メモリへのアクセスが必要となる。このため、レジューム時から通常の処理動作時に移行する際には、再度外部メモリを立ち上げる必要があるため、迅速な処理ができないとともに、レジューム時にのみ外部メモリへの電力供給を停止するため、十分に消費電力の低減を図ることができないという問題がある。

【 0 0 1 0 】

すなわち、上記従来の情報処理装置では、レジュームモードへの移行の際に、復帰用のデータの退避や、復帰時の外部メモリが使用可能であるか否かのチェックを目的としてキャッシュを使用している。よって、何れの情報処理装置においても、タスクの処理等の動作は、ダイナミック素子や外部メモリの復帰後に開始される。したがって、レジューム時以外には常に外部メモリが起動していることになるため、十分に消費電力の低減を図ることができない。

【 0 0 1 1 】

本発明は、上記の問題点に鑑みてなされたものであり、その目的は、外部メモリを使用せずに CPU 内部で処理を行うことで、迅速な処理が可能になるとともに、外部メモリに対する電力供給を停止し、消費電力を低減することができる C

P Uおよびこれを備えた情報処理装置、C P Uの制御方法を提供することにある。

【0012】

【課題を解決するための手段】

本発明のC P Uは、上記の課題を解決するために、内部にキャッシュを備えており、上記キャッシュに対してデータの書き換えを行うとともに、この書き換えたデータを所望のタイミングで外部メモリに反映させるライトバック制御を行うC P Uにおいて、処理するタスクの必要メモリ量に応じて、上記キャッシュだけにアクセスしてタスクの処理が可能であるか否かを判定し、該処理が可能である場合には、上記外部メモリに対するアクセスを禁止する制御手段を備えていることを特徴としている。

【0013】

上記の構成によれば、C P U内に備えられているキャッシュだけを用いてタスクの処理を行うことができる状態においては、外部メモリへのアクセスを禁止しているため、外部メモリに対する電力供給を停止させて消費電力を削減できる。

【0014】

すなわち、通常、ライトバック制御を行うC P Uでは、所定のタイミングで、データ等が書き込まれたキャッシュから外部メモリに対して該データ等の書き込みを行うが、本発明のC P Uでは、一定条件下において、キャッシュへのアクセスが可能な状態のまま、外部メモリへのアクセスを禁止している。

【0015】

例えば、処理するタスクの必要メモリ量が比較的小さく、キャッシュの残り容量だけでタスクの処理を継続させることが可能である場合には、外部メモリに対するアクセスを禁止するとともに、キャッシュだけにアクセスしてタスクの処理を行う。

【0016】

このように、キャッシュだけにアクセスしてタスクの処理が可能な状態においては、外部メモリへのアクセスを禁止しているため、アクセス禁止期間中には、外部メモリに対する電力供給を停止する、あるいは低消費電力モードに切り替え

る等の措置を取ることができ、結果として、本発明のCPUを備えた情報処理装置の消費電力を削減することができる。

【0017】

また、本発明のCPUでは、以上のように、キャッシュのみで行える処理は外部メモリを使用せずに処理を行い、できる限り外部メモリを使用しないようにすることで、外部メモリに供給する電力消費量を削減している。これにより、レジューム時にのみ外部メモリに対する電力供給を停止させる従来の情報処理装置と比較して、外部メモリに対する電力供給停止期間を長くでき、より効果的に消費電力を削減できる。

【0018】

さらに、キャッシュよりも低速な外部メモリに対するアクセス回数を従来よりも減らし、高速アクセスが可能なキャッシュを用いて処理を行うため、迅速な処理が可能になる。

【0019】

なお、上記キャッシュは、プログラムコード等を記憶している命令キャッシュ、データ等を記憶しているデータキャッシュおよびこれらを補完するために設けられている2次キャッシュ等を含んでいる。

【0020】

上記制御手段は、上記キャッシュの残り容量および／または処理するタスクの必要メモリ量を検出することがより好ましい。

【0021】

これにより、例えば、キャッシュの残り容量とタスクの必要メモリ量とを検出して両者を比較することで、キャッシュのみを用いて該タスクの処理を行うことが可能であるか否かを判定できる。

【0022】

このとき、判定の結果が「可能」であった場合には、外部メモリに対するアクセスを禁止するとともに電力供給を停止する。一方、判定の結果が「不可能」であった場合には、そのまま外部メモリにアクセスしてライトバック制御を行い、キャッシュの残り容量を増やすことができる。

【0023】

また、キャッシュの残り容量が殆どないことが検出された場合には、タスクの必要メモリ量を検出することなく、外部メモリにアクセスしてライトバック制御を行い、キャッシュの残り容量を増やすことができる。

【0024】

上記外部メモリに対するアクセスが禁止されている状態において、上記制御手段が、上記キャッシュだけでは処理が不可能と判断した場合、あるいはキャッシュミスが発生した場合には、上記外部メモリに対するアクセス禁止を解除することがより好ましい。

【0025】

これにより、外部メモリに対するアクセス禁止状態において、アクセス禁止を解除することで、ライトバック制御を行って、キャッシュ内のデータ等を外部メモリに書き込むことができるため、外部メモリに書き込まれたデータ等をキャッシュから消去して、キャッシュの残り容量を増加させることができる。

【0026】

そして、キャッシュの残り容量がタスクの処理に十分な容量まで増加した場合、あるいは外部メモリからキャッシュに必要なデータ等が書き込まれた後には、再度、キャッシュだけでタスクの処理を行うとともに、外部メモリに対するアクセスを禁止して消費電力を削減することが可能になる。

【0027】

また、このようなキャッシュだけにアクセスして処理を行うモードから、キャッシュに加えて外部メモリに対してもアクセスして処理を行うモードへの移行を、処理を停止させることなく、処理を継続したまま行うことができたため、迅速な処理が可能になる。

【0028】

なお、上記キャッシュミスとは、読みに行ったデータがキャッシュ内に存在しなかった場合をいう。

【0029】

内部クロックのクロック周波数を制御するクロック制御手段を備えており、該

クロック制御手段は、上記外部メモリに対するアクセスが禁止された場合には、上記クロック周波数を変化させることがより好ましい。

【0030】

これにより、例えば、外部メモリへのアクセスが禁止されている場合、すなわち、高速アクセスが可能なキャッシュのみにアクセスして処理を行っている場合には、クロック周波数を低くすることにより、CPUの消費電力を削減することができる。

【0031】

なお、このようにクロック周波数を低くした場合には処理速度が低下することが考えられるが、キャッシュは高速アクセスが可能であるため、処理速度をそれほど落とさずに処理を行うことが可能である。

【0032】

また、キャッシュのみで処理を行う場合において、高速アクセスが可能なキャッシュを用いているため、クロック周波数を高めることで、さらに高速処理が可能になる。

【0033】

上記制御手段は、上記キャッシュ内における不要なデータを格納しているアドレスを検出し、該検出されたアドレスに対応するキャッシュ領域を解放することがより好ましい。

【0034】

これにより、例えば、処理の終了したタスク専用のプログラム、データの領域、スタックの不要なデータ等を格納しているアドレスを検出し、このデータ等に対応するキャッシュ領域を、データが書き込まれていない領域とみなして解放することにより、実質的にキャッシュの空き容量を増やすことができるため、キャッシュを有効に利用できる。

【0035】

さらに、キャッシュ内の不要になったデータ等については、外部メモリへのライトバックを行う必要はないため、上記のように該データ等のメモリ領域を開放することで、無駄なライトバックが行われることを防止できる。

【 0 0 3 6 】

電源投入後の初期動作時においては、所望のプログラムおよびデータを上記外部メモリから上記キャッシュに読み込んだ上で、上記外部メモリへのアクセスを禁止することがより好ましい。

【 0 0 3 7 】

これにより、初期動作時に、外部メモリから所望のプログラムおよびデータを読み込んでいるため、キャッシュには動作に必要な情報が読み込まれているとともに、残り容量が比較的多い状態となっている。

【 0 0 3 8 】

よって、CPUが備えているキャッシュのみにアクセスして処理を行うことができるとともに、外部メモリへのアクセスを禁止して外部メモリに対する電力供給を停止することで、初期動作時から消費電力を削減した状態にできる。

【 0 0 3 9 】

上記制御手段は、タスクの状況が変化する際に、上記外部メモリに対するアクセスが必要か否かについての判断を行うことがより好ましい。

【 0 0 4 0 】

これにより、OS等によって管理されているタスクの状況が変化した場合には、外部メモリへのアクセスが必要になる可能性が生じるため、その時点で外部メモリを使用する必要があるか否かについて判断し、外部メモリが必要な場合には、実際にアクセスが発生する前に外部メモリを起動することができるため、外部メモリの起動待ち時間を短縮して、処理を迅速に行うことができる。

【 0 0 4 1 】

また、タスクの状況の変化に応じて判断するため、オーバヘッドを大きくしなくても、外部メモリへのアクセスが必要であるか否かの判断が可能になる。

【 0 0 4 2 】

さらに、例えば、一定期間ごとに外部メモリが必要か否かのチェックを行う場合と比較して、外部メモリへのアクセスが可能であるか否かのチェックの回数を少なくすることができる。

【 0 0 4 3 】

上記制御手段は、上記キャッシュ内のプログラムおよびデータについて、パージされたか否かについて判定し、パージされていない場合には、上記プログラムおよびデータを上記外部メモリから読み込む処理を行わないことがより好ましい。

【0 0 4 4】

これにより、キャッシュ内に書き込まれたプログラムやデータについて、これらがパージされていない場合には、該プログラム等はキャッシュ内に残っているため、キャッシュのみで処理を行うモードに移行する場合に必要なプログラム等を、新たに外部メモリから読み込む処理を省略して、無駄な処理を減らすことができる。

【0 0 4 5】

すなわち、外部メモリへのアクセスを禁止してキャッシュだけで動作するモードへ移行する際には、外部メモリからキャッシュに対して、キャッシュだけで動作するために必要なプログラム等をキャッシュへ書き込む必要がある。しかし、当該プログラム等がパージされていない場合には、既にキャッシュ内に存在しているため、再度外部メモリから同じプログラム等を読み出す必要はない。

【0 0 4 6】

そこで、本発明のCPUは、キャッシュ内のプログラム等がパージされたか否かを判定し、必要なプログラム等だけを外部メモリから読み出すことで、外部メモリに対する不要なアクセスを省くことができる。

【0 0 4 7】

本発明の情報処理装置は、上記の課題を解決するために、上記CPUと、外部メモリと、該外部メモリへ電力を供給する電源供給手段とを備えた情報処理装置において、上記電源供給手段は、上記外部メモリへのアクセスが禁止されている場合には、該外部メモリへの電力供給を停止することを特徴としている。

【0 0 4 8】

上記の構成によれば、外部メモリへのアクセスが禁止されている場合、すなわちCPUが内部に備えているキャッシュに対してアクセスするだけで処理を継続できる場合には、外部メモリへの電力供給を停止することで、情報処理装置の消

費電力を削減できる。

【0049】

上記外部メモリは、複数のモジュールを含んでおり、上記制御手段は、上記モジュールごとに電力供給の制御を行うことがより好ましい。

【0050】

これにより、外部メモリが、例えば、RAM、ROM等の複数のモジュールからなる場合には、このモジュールの性質に応じて電力供給を制御することで、さらに効率的に情報処理装置の消費電力を低減できる。

【0051】

例えば、キャッシュのみにアクセスして処理を行うモードに移行する際の、プログラムやデータをキャッシュに読み込む場合には、ROMだけを起動して電力供給を行う。一方、処理中にデータが満杯になった場合には、RAMだけを起動して電力供給を行う。

【0052】

このように、各モジュールの性質および状況に応じて、外部メモリに対する電力供給を細かく制御することで、より効果的に情報処理装置の消費電力を削減することができる。

【0053】

本発明のCPUの制御方法は、上記の課題を解決するために、内部に備えたキャッシュに対してデータの書き換えを行うとともに、この書き換えたデータを所望のタイミングで外部メモリに反映させるライトバック制御を行うCPUの制御方法において、処理するタスクの必要メモリ量に応じて、上記キャッシュだけにアクセスしてタスクの処理が可能であるか否かを判定し、該処理が可能であると判定した場合には、上記外部メモリに対するアクセスを禁止することを特徴としている。

【0054】

上記の制御方法によれば、CPU内に備えられているキャッシュだけを用いてタスクの処理を行うことができる状態においては、外部メモリへのアクセスを禁止しているため、外部メモリに対する電力供給を停止させて消費電力を削減でき

る。

【0055】

すなわち、通常、ライトバック制御を行うCPUでは、所定のタイミングで、データ等が書き込まれたキャッシュから外部メモリに対して該データ等の書き込みを行うが、本発明のCPUの制御方法では、一定条件下において、キャッシュへのアクセスが可能な状態のまま、外部メモリへのアクセスを禁止している。

【0056】

例えば、処理するタスクの必要メモリ量が比較的小さく、キャッシュの残り容量だけでタスクの処理を継続させることが可能である場合には、外部メモリに対するアクセスを禁止するとともに、キャッシュだけにアクセスしてタスクの処理を行う。

【0057】

このように、キャッシュへのアクセスだけでタスクの処理が可能な状態においては、外部メモリへのアクセスを禁止しているため、アクセス禁止期間中には、外部メモリに対する電力供給を停止する、あるいは低消費電力モードに切り替える等の措置を取ることができる。よって、結果として、本発明のCPUの制御方法を採用した情報処理装置の消費電力を削減することができる。

【0058】

また、本発明のCPUの制御方法では、以上のように、キャッシュのみで行える処理は外部メモリを使用せずに処理を行い、できる限り外部メモリを使用しないようにすることで、外部メモリに供給する電力消費量を削減している。これにより、レジューム時にのみ外部メモリに対する電力供給を停止させる従来の情報処理装置と比較して、外部メモリに対する電力供給停止期間を長くでき、より効果的に消費電力を削減できる。

【0059】

さらに、キャッシュよりも低速な外部メモリに対するアクセス回数を従来よりも減らし、高速アクセスが可能なキャッシュを用いて処理を行うため、迅速な処理が可能になる。

【0060】

【発明の実施の形態】

本発明のCPUおよびこれを備えた情報処理装置、CPUの制御方法の一実施形態について、図1～図7に基づいて説明すれば以下の通りである。

【0061】

PC (Personal Computer、情報処理装置) 1は、図2に示すように、CPU (Central Processing Unit) 10、電源20、外部ROM (Read Only Memory) (外部メモリ、モジュール) 30、および外部RAM (Random Access Memory) (外部メモリ、モジュール) 40を備えている。

【0062】

これらの各部材は、システムバス (System-bus) 50を介して接続されている。また、PC 1は、さらに図示しない入力部としてのキーボード、および表示部としてのディスプレイを備えている。

【0063】

CPU 10は、PC 1の動作制御を行う。なお、CPU 10の詳細な構成については、図1を用いて後段にて詳述する。

【0064】

電源 (電源供給手段) 20は、PC 1の本体およびPC 1が備えている各部材への電力供給を行うものであり、電源管理部21を備えている。

【0065】

電源管理部21は、電力供給線22を介して各部材に対して電力を供給しており、CPU 10からの制御信号に応じて、外部ROM 30および外部RAM 40に対する電力供給を停止したり、再開したりする。

【0066】

外部ROM 30および外部RAM 40は、CPU 10の外部に備えられた、CPU 10の動作に必要な情報等を格納するための記憶装置である。

【0067】

外部ROM 30は、固定的にデータを保持できる一方、CPU 10からの新たな書き込みはできない記憶装置である。

【0068】

外部RAMは40、CPU10からの書き込みは可能であるが、固定的にデータを保持することはできない記憶装置である。

【0069】

CPU10は、このような両記憶装置の性質の違いによって、適宜、外部ROM30と外部RAM40とを使い分けて使用している。

【0070】

例えば、外部ROM30には、図2に示すように、プログラムのデータとパラメータ（固定データ）とが記録されている。また、外部RAM40には、CPU10における処理中に、グローバル変数、ローカル変数、スタックデータなどが記録される。

【0071】

ここで、CPU10の詳細な構成について、図1を用いて以下で説明する。

【0072】

CPU10は、図1に示すように、バスインターフェース11、制御ユニット12、命令キャッシュ（キャッシュ）13a、データキャッシュ（キャッシュ）13b、2次キャッシュ（キャッシュ）13c、命令デコーダ14、演算ユニット15およびレジスタ群16を備えている。

【0073】

このCPU10は、図2に示す外部メモリとしての外部ROM30または外部RAM40からバスインターフェース11を介してデータを取得し、プログラムの命令を解読して、演算またはデータ転送などの命令を実行する。

【0074】

バスインターフェース11は、データバス11a、アドレスバス11b、制御信号（線）11cからなる、図2に示すシステムバス50によって、データのやりとりを行う。

【0075】

制御ユニット12は、CPU10が備えている各部材の動作制御を行っており、CPU10において処理するタスクの必要メモリ量を見積もることができる。また、制御ユニット12は、内部クロック制御部（クロック制御手段）17、M

MU (Memory Management Unit：メモリ管理装置) 1 8、Mフラグ 1 9 a、Pフラグ 1 9 bを備えている。

【0 0 7 6】

内部クロック制御部 1 7は、クロックを生成し、CPU 1 0の全体の処理をこのクロックに従って行わせており、生成するクロック周波数を変化させることができる。

【0 0 7 7】

MMU 1 8は、命令に応じてメモリのアドレスを生成、供給するものである。

【0 0 7 8】

Mフラグ 1 9 a、Pフラグ 1 9 bについては、後段にて詳述する。

【0 0 7 9】

命令デコーダ 1 4は、CPU 1 0内で発生した命令(コード)を解読するためのものであり、解読後、制御ユニット 1 2、演算ユニット 1 5等によって、データのやり取りや演算が行われる。なお、ここでのデータのやり取りについては、必要なデータが外部メモリ(外部ROM 3 0、外部RAM 4 0)に記録されている場合もある。

【0 0 8 0】

命令キャッシュ 1 3 aおよびデータキャッシュ 1 3 bは、CPU 1 0の内部に備えられている高速アクセスが可能な記憶装置である。

【0 0 8 1】

命令キャッシュ 1 3 aは、プログラムのコードを格納するためのキャッシュである。CPU 1 0において実行される命令は、命令キャッシュ 1 3 aに格納された外部メモリに格納されたプログラムが、命令キャッシュ 1 3 aから読み出し、解読されて実行される。

【0 0 8 2】

データキャッシュ 1 3 bは、CPU 1 0が命令を実行する際に処理の対象となる変数が格納されるキャッシュである。

【0 0 8 3】

この命令キャッシュ 1 3 aおよびデータキャッシュ 1 3 bは、一般的に、1次

キャッシュと呼ばれ、例えば、16 Kバイト程度の格納領域を有している。命令キャッシュ13aおよびデータキャッシュ13bには、キャッシュ毎に、キャッシュ上における変数の格納場所と外部メモリのアドレスを対応づける図示しないTAGや、タスク（プロセス）ごとの論理アドレスと物理アドレスとを対応づけるための、TLB（Translation Look-aside Buffer）が備えられている。

【0084】

2次キャッシュ13cは、1次キャッシュ（命令キャッシュ13a、データキャッシュ13b）を補完するために備えられており、1次キャッシュの容量が満杯の場合に1次キャッシュに入りきらないデータ等を格納する、1次キャッシュよりも低速であるが、外部メモリよりも高速アクセスが可能な記憶装置である。

【0085】

このように、命令キャッシュ13aおよびデータキャッシュ13bを補完する2次キャッシュ13cを備えていることで、キャッシュ13a～13cの記憶容量を増大させ、より外部メモリを使用する回数を減らすことができる。

【0086】

レジスタ群16は、命令を実行する際にデータが格納される記憶装置であって、プログラムによって操作参照可能なものと、演算ユニット15が処理中のデータを格納する場合のように直接操作できないものがある。

【0087】

プログラムによって操作参照可能なレジスタには、処理の対象となる変数やアドレスなどを格納する汎用レジスタと、スタックポインタ（以下、SPと示す）のような用途が特定されたレジスタとがある。

【0088】

SPは、主として、サブルーチンコール（関数呼び出し）の際の引数や戻り番地などを格納するために利用されるスタック領域の使用済み領域と未使用領域の境界のアドレスを保持している。

【0089】

次に、以上のような構成のCPU10による処理について、図3（a）および図3（b）を用いて説明する。

【0090】

なお、図3（a）には、これらキャッシュの使用状況の時間変化に応じたMフラグ、Pフラグの状況についても示している。

【0091】

まず、Mフラグ、Pフラグについて説明する。

【0092】

Mフラグは、外部メモリへのアクセスが許可されているか、禁止されているかを示すフラグであって、本実施形態においては、Mフラグの値が「1」の場合には、外部メモリにアクセスして情報を反映させることが可能であることを指し、一方、Mフラグの値が「0」の場合には、外部メモリへのアクセスが禁止されたことを指す。

【0093】

このように、制御ユニットは、Mフラグを用いて、外部メモリ（外部ROM30、外部RAM40）へのアクセス禁止、解除を行う。

【0094】

例えば、キャッシュ13a～13cが一杯でデータを書き込む余裕がない場合や、キャッシュ13a～13cに参照すべきデータが無い場合には、Mフラグ「1」を入力して外部メモリへのアクセスを行う。また、処理によってはメモリが大量に必要となるため、始めから外部メモリを立ち上げておけばよい。なお、Mフラグの値の設定処理については後段にて説明する。

【0095】

本実施形態においては、例えば、簡単な処理のみを行う場合には、初めに必要な情報をキャッシュ13a～13cに読み込むことによって、外部メモリを不要にし、外部メモリへのアクセスを禁止できる。なお、これらのキャッシュのデータは、最終的には必要に応じて外部メモリへと書き出される。

【0096】

Pフラグは、キャッシュ13a～13cに備えられているデータが破壊されているか否かを判定するためのフラグである。具体的には、Pフラグが「0」の場合にはキャッシュ13a～13c内のデータが破壊されていないことを示し、「

1」の場合にはデータが破壊されていることを示す。

【0097】

例えば、キャッシュ13a～13cの容量が満杯となり、キャッシュ13a～13cに記録されている情報を外部RAM40に退避させて他のデータ等を上書きする場合には、所定の時点でこの退避させた情報をキャッシュ13a～13cに記録しなおす必要がある。また、外部RAM40に退避させなくとも、例えば、外部ROM30に記録されているデータの場合には、単にキャッシュ13a～13cの中のデータをパージしてもよい。このような場合には、データ等が破壊されているため、外部ROM30あるいは外部RAM40からのデータ等の再読み込みが必要となる。

このように、データ等の破壊、パージの有無および再読み込みの要否について、Pフラグによって判定できる。すなわち、データの破壊がなされた場合には、Pフラグの値を「1」とし、これに応じて再読み込みを行う。一方、データの破壊がなされていない場合には、Pフラグの値を「0」とし、そのまま処理を行う。

。

【0098】

ここで、命令キャッシュ13aとデータキャッシュ13bとMフラグ、Pフラグとを用いた実際のCPU10の動作について、図3(a)、図3(b)を用いて説明する。

【0099】

命令キャッシュ13aは、図3(a)に示すように、命令キャッシュ13a中で使用されている領域における占有面積に応じて幅を変えてタスクごとに示している。

【0100】

また、データキャッシュ13bについては、データキャッシュ13b中で使用されている領域における占有面積に応じて幅を変えて変数ごとに示している。

【0101】

ここで、図3（a）に示すグローバル変数とは、いろいろなタスクにおいて共通して用いられる変数を意味する。ここでは、TLB等のシステムが利用する領域も確保されているものとする。一方、ローカル変数とは、一つのタスク毎に用いられる変数を意味する。これらの変数は、それぞれのキャッシュのメモリ領域中において固定的に割り当てられる。

【0102】

また、図3（a）に示すスタックとは、関数などの実行中に、処理に応じてテンポラリに、すなわち動的に、メモリ領域が割り当てられる領域に相当する。なお、このスタックの領域は、タスクごとに割り当てられるメモリ領域について、模式的に全タスク分に相当する領域を示したものである。

【0103】

また、図3（b）は、図3（a）に続く状態の一例を示したものである。

【0104】

図3（a）、図3（b）に示すように、一つのプログラム（タスク）を実行する場合に、どの部分が命令キャッシュ13aに読み込まれるかということは、予め決められている。また、プログラム中のどの変数がデータキャッシュ13bに読み込まれるかということも、予め決められている。

【0105】

本実施形態のCPU10は、図3（a）に示すように、起動の直後には、タスク0～2が命令キャッシュに読み込まれる。また、これらのタスクに対応する変数が、データキャッシュに読み込まれる。なお、上記タスク0～2は、例えば、画面の制御、通信の制御といったCPU10の基本的な処理を行うものである。

【0106】

一方、その後に起動されるタスク3・4は、基本的な処理以外の、何らかの付加的な処理を行うものである。

【0107】

タスク3の起動要求に応じて、Mフラグが「1」となる。これは、タスク3の処理を行うために必要なコード、データ等を外部メモリから読み込むためである。

。読み込みによって、命令キャッシュ 13 a とデータキャッシュ 13 b とにそれぞれメモリ領域が確保される。そして、読み込み完了後には、Mフラグが「0」となる。タスク 3 が終了すると、命令キャッシュ 13 a とデータキャッシュ 13 b とに確保されていたメモリ領域が解放される。

【0108】

また、タスク 4 の起動要求の際にも、同様に Mフラグが「1」となってデータ等の読み込みが行われ、その後、Mフラグが「0」となる。

【0109】

ここで、図 3 (a) において斜線で示したように、タイミング T 1 において、各タスクによるスタック使用量が、データキャッシュ 13 b の記憶容量を越えてしまい、容量不足となる場合がある。この場合には、Mフラグを「1」とし、スタックに記憶されていた情報を外部メモリなどに退避させる。すなわち、例えば、外部 RAM 40 を用いて、各キャッシュ 13 a ～ 13 c と外部メモリとを一体のメモリとして用いるようにする。なお、この場合には、Pフラグを「1」として、このような退避が行われたことを判定可能とする。

【0110】

この後、タスク 4 の処理が終了すると、図 3 (b) に示すように、命令キャッシュ 13 a とデータキャッシュ 13 b とに確保されていた、タスク 4 に対応する領域、変数領域およびスタック領域は解放される。これによって、退避されていた情報の再読み込みがなされて、起動後と同様に、Mフラグと Pフラグとがともに「0」となる。

【0111】

また、起動後から実行されているタスク 0 ～ 2 において、あるタイミングにおいてタスク 2 の実行が本格化すると、図 3 (b) の斜線部として示すように、タイミング T 2 において、タスク 2 によるプログラムおよびデータの使用の増加が見込まれる。この場合には、Mフラグを「1」として、スタックに記憶されていた情報を外部メモリなどに退避させ、キャッシュ 13 a ～ 13 c と外部メモリとを一体のメモリとして用いるようにする。また、Pフラグを「1」として、このようなキャッシュ 13 a ～ 13 c のコード、データ等が破壊されたことを認識可

能とする。

【0112】

そして、タイミングT3にてタスク2の実行が通常に戻ると、命令キャッシュとデータキャッシュとに確保されていた、タスク2のプログラム領域、変数領域およびスタック領域は解放される。

【0113】

これにより、外部メモリからキャッシュ13a～13cへ、退避させた情報の再読み込みを行い、その後、起動直後と同様に、MフラグとPフラグとがともに「0」となる。

【0114】

以上のように、本実施形態のCPU10は、外部メモリへのアクセスが必要な場合には、Mフラグを「1」として外部メモリへのアクセスを行うことで、処理を継続したまま外部メモリへのアクセスを可能とし、キャッシュ13a～13cと外部メモリとを一体のメモリとして用いることができる。

【0115】

さらに、このような外部メモリへのアクセスにより、キャッシュ13a～13cからデータ等が退避された場合には、Pフラグを「1」とすることで、キャッシュ13a～13c内のデータ等が破壊されていることを認識できる。これにより、タスクが終了してキャッシュ13a～13c内に空き容量ができた場合には、そのPフラグに対応するデータ等について再読み込みを行うことで、再度キャッシュ13a～13cのみを用いた処理が可能になる。

【0116】

次に、データキャッシュ13b内におけるスタック領域の使用状況について、図4(a)～図4(e)を用いて説明する。

【0117】

なお、スタック領域は、タスクごとに、図4(a)～図4(e)に示すような使用状態となっており、ボトムからSP(Stack Pointer:スタックポインタ)までが各タスクに対応する使用領域に相当する。

【0118】

タスク開始前またはタスク終了後においては、図 4（a）に示すように、使用領域は少ないため、S P が低い位置にある。そして、タスクの処理の進行に従って、順次スタック領域が使用されて S P の位置が上昇し、図 4（b）～図 4（e）に示すような状態となる。

【0 1 1 9】

タスクが終了した場合には、図 4（f）に示すプログラムに従って、図 4（e）の状態から順次スタック領域が解放され、図 4（a）の状態に戻るようになっているが、これについては後段にて詳述する。

【0 1 2 0】

ここでさらに、本実施形態の C P U 1 0 による処理について、図 5（a）～図 7（b）に示すフローチャートに基づいて説明すれば、以下の通りである。

【0 1 2 1】

まず、C P U 1 0 による、外部メモリに対するアクセスを禁止する処理について説明する。

【0 1 2 2】

なお、以下に示す処理は、C P U 1 0 の起動時、タスクの処理が終了した後、または必要なデータを外部メモリから読み出した後、例えば、図 3 に示すタスク 3 の起動要求によって読み込んだ後、などに行われる。

【0 1 2 3】

ステップ（以下、S と示す）1 においては、動作をリセットするか否かを判定する。例えば、C P U 1 0 の起動時には動作をリセットして S 3 に進む。リセットしない場合には S 2 に進む。

【0 1 2 4】

S 2 においては、P フラグが「0」であるか（オフであるか）否かが判定され、例えば、図 3 に示すタスク 3 の起動要求時には、P フラグは「0」であるため S 7 に進む。また、例えば、図 3 に示すタイミング T 3 のような、斜線状態の終了後には、P フラグは「1」であるため S 3 に進む。

【0 1 2 5】

S 3 においては、S 2 において P フラグが「1」と判定されているので、例え

ば、キャッシュ 13 a～13 c から退避させたプログラム、あるいは単にキャッシュ 13 a～13 c から消去した所定のプログラムについて、外部 ROM 30 から命令キャッシュに取り込む。

【0126】

S 4 においては、読み込んだプログラムに対応するデータをデータキャッシュ 13 b へ取り込む。

【0127】

S 5 においては、データキャッシュ 13 b に変数領域の割り当てを行う。

【0128】

なお、S 3～S 5 において、必要なデータ等を再読み込みした場合には、S 6 において P フラグを「0」に戻す。

【0129】

S 7 においては、S 1～S 6 までの処理により、必要なデータがキャッシュ 13 a～13 c の内部にある、または外部メモリからキャッシュ 13 a～13 c に再読み込みが行われた状態であるため、M フラグを「0」に戻し、S 8 において外部メモリに対するアクセスを禁止し、処理を終了する。

【0130】

このように、外部メモリに対するアクセスを禁止し、その後の処理をキャッシュ 13 a～13 c のみを用いて行うことで、外部メモリへの電力供給を停止することができるため、PC 1 の消費電力を削減できる。なお、外部 RAM 40 については、データを保持したまま低消費電力モードへ移行することで、PC 1 の消費電力を削減するようにしてもよい。

【0131】

なお、キャッシュ 13 a～13 c のみを用いて処理を行う場合には、高速処理が可能であるため、上記内部クロック制御部 17 のクロック周波数を低下させた場合でもある程度の処理速度が得られる。このため、クロック周波数を低下することで、CPU 10 の消費電力を低減できる。逆に、キャッシュ 13 a～13 c のみを用いて処理を行う場合において、クロック周波数を高くすることにより、より高速処理が可能になる。

【0132】

また、このようなCPU10の起動時（初期動作時）においては、所望のプログラムおよびデータを外部メモリからキャッシュ13a～13cに読み込んだ上で、外部メモリへのアクセスを禁止することが好ましい。

【0133】

これにより、初期動作時においては、キャッシュ13a～13cのみを用いた処理が可能になり、立ち上げ時から消費電力を削減できる。

【0134】

次に、図5（b）を用いて、本実施形態のCPU10によるタスク開始時の処理について説明する。

【0135】

S9においては、今から処理を開始するタスクの処理に、外部メモリが必要であるか否かについて判定する。このような、外部メモリの要否については、それぞれのタスクごとに個別に判断される。

【0136】

例えば、図3（a）に示すタスク0～2においては、キャッシュ13a～13cのみで処理が可能であり、外部メモリは不要であるので、S11へ進む。

【0137】

一方、図3（a）に示すタスク4や、図3（b）に示すタスク2では、処理が本格化してキャッシュ13a～13cの残り容量が少なくなり、外部メモリが必要となっているため、S10へ進み、S10において外部メモリが起動される。

【0138】

S11においては、タスクの処理が開始される。

【0139】

S12においては、その時点におけるキャッシュ13a～13cの使用状態によって、キャッシュ13a～13cの残りの領域のみで十分か、または外部メモリが必要であるか否かについて判定される。

【0140】

例えば、図3（a）に示すタスク3については、外部メモリが不要であるため

、S13に進み、図5（a）に示す外部メモリアクセス禁止モードに移行した後、処理を終了する。一方、例えば、図3（a）に示すタスク4のように、処理途中においてキャッシュ13a～13cの残り容量が少なくなり、外部メモリが必要となった場合には、図5（a）に示す外部メモリアクセス禁止モードへ移行することなく、そのまま処理を終了する。

【0141】

次に、図5（c）を用いて、外部メモリを初期化する処理について説明する。

【0142】

なお、図5（b）のS10における外部メモリ起動の動作は、外部メモリへ起動指示を行い、図5（c）に示す外部メモリ初期化完了の割り込みを受けることを含んでいるものとする。

【0143】

本実施形態のCPU10は、図5（c）に示すように、この割り込みを受けた後に、S14においてMフラグを「1」にしてから、図5（b）のS11においてタスク処理を開始する。これは、S10において外部メモリへの起動指示を行った場合でも、外部メモリが直ぐに初期化されて使用可能となる訳ではなく、例えば、リフレッシュ処理のような初期化処理が必要となるからである。したがって、CPU10は、初期化完了の割り込みを受けた後、Mフラグを「1」として

【0144】

なお、CPU10は、自分のプログラムを実行するのみではなく、外部からの処理要求（刺激）（割り込み）に応じて処理をするような仕組みを持っている。例えば、キーボードへの入力、CPU10への割り込み指示として実現されている場合には、CPU10によるキーボードの監視が不要となり、処理が容易になる。

【0145】

ここで、キャッシュ13a～13c内にヒープ領域を確保した場合の処理について、図5（d）を用いて説明する。

【0146】

まず、キャッシュ 13 a～13 c にヒープ領域を確保する場合には、S 15 において、キャッシュ 13 a～13 c の容量が超過しているか否かについて判定する。ここで、超過していない場合には S 17 へ進み、超過している場合には、S 16 において外部メモリを起動する。そして、S 17 においては、メモリアロケートを行い、ヒープ領域を確保した後、処理を終了する。

【0147】

このように、タスクの処理中において、キャッシュ 13 a～13 c 内にヒープ領域を確保する場合には、キャッシュ 13 a～13 c の容量が超過状態となっているか否かについて判定する。ここで、超過状態である場合には、外部メモリへのアクセスが発生する前に外部メモリを起動することで、外部メモリの立ち上がり待ち時間を短縮できる。

【0148】

さらに、あるタスクの処理中に別のタスクの処理が開始された場合の処理について、図 5（e）を用いて説明する。

【0149】

図 5（e）に示すように、別タスクの処理が開始されると、S 18 において子タスクが生成され、S 19 において別タスクの処理が本格化したか否かについて判定する。

【0150】

ここで、処理が本格化していない場合には、図 5（b）に示す単独のタスクを処理する工程へ進む。一方、S 19 において別タスクの処理が本格化した場合には、この時点において外部メモリを起動する。そして、S 20 において外部メモリを起動した後、S 21 においてタスクの処理を開始する。

【0151】

これにより、外部メモリが必要となってから外部メモリを立ち上げる場合と比較して、事前に外部メモリを立ち上げることができるため、立ち上がりまでの待ち時間を短縮して迅速な処理が可能になる。

【0152】

また、図 5（d）、図 5（e）に示すように、タスクの状況が変化する際に、

外部メモリへのアクセスが必要か否かの判断を行うことで、外部メモリへのアクセスが必要になってから起動する場合と比較して、事前に外部メモリを起動することができるため、外部メモリへのアクセスを迅速に行うことができる。

【0 1 5 3】

次に、図 4（e）に示すように、空き容量が少なくなったデータキャッシュ 1 3 b を解放する際の処理について、図 6（a）を用いて説明する。

【0 1 5 4】

なお、この処理は、上述した図 4（e）～図 4（a）に示した、データキャッシュ 1 3 b 内におけるスタック領域に関する関数呼び出しからの戻り動作に相当する。

【0 1 5 5】

また、以下では、説明の便宜上、図 1 に示した 2 次キャッシュ 1 3 c がいないものとして説明する。実際の処理においては、データキャッシュ 1 3 b の空き容量が少なくなった場合には、2 次キャッシュ 1 3 c に必要なデータ等を格納して処理を行うため、データキャッシュ 1 3 b のみが容量超過したからといって、外部メモリへのアクセス禁止を解除するものではない。

【0 1 5 6】

ここで、データキャッシュ 1 3 b は、図 4（e）に示すように、ほとんど空き容量がない状態となっているものとする。

【0 1 5 7】

S 3 0 においては、次の命令が、B P（Base Pointer）を S P とするという内容であるか否かについて判定し、データキャッシュ 1 3 b 内を解放する方向に処理が進んでいることを確認する。

【0 1 5 8】

ここで、「N o」である場合には再び S 3 0 へ戻り、「Y e s」の場合にはデータキャッシュ 1 3 b 内を解放する方向に処理が進んでいるものと認識し、B P を S P とする処理を行って、図 4（d）に示すような状態となる。

【0 1 5 9】

S 3 1 においては、S P の値（x）を所定のレジスタに退避させ、S 3 2 にお

いて、次の命令が「RETURN」であるか否かを判定することで、データキャッシュ 1 3 b 内を解放する方向に処理が進んでいるか否かを判定する。

【0 1 6 0】

ここで、次の命令が「RETURN」である場合には、図 4 (c) に示す「戻り番地」を除去した状態となる。一方、次の命令が「RETURN」でない場合には、戻り動作ではないと判断して S 3 2 に戻る。

【0 1 6 1】

S 3 3 においては、次の命令が S P に関係するものであるか否かについて判定し、関係する場合には S 3 3 を繰り返す。一方、S P に関係しない場合には、S 3 4 に進んで、S P の値が先ほど退避した (X) の値よりも大きいか否かについて判定する。

【0 1 6 2】

ここで、S P の値が (x) よりも小さい場合、例えば、戻った後すぐに別の関数を呼び出した場合等、においては、S 3 0 に戻る。一方、S P の値が (x) よりも大きい場合には、データキャッシュ 1 3 b 内に十分な空き容量が形成されたことが認識できるため、S 3 5 へ進む。

【0 1 6 3】

S 3 5 においては、データキャッシュ 1 3 b のスタック領域の S P 未満のアドレスを、未書き込み状態に変更する。

【0 1 6 4】

これにより、データキャッシュ 1 3 b 内の領域を、図 4 (c) に示す状態から、図 4 (b) に示す状態を経て、図 4 (a) に示すような解放された状態にすることができる。

【0 1 6 5】

続いて、S 3 6 において、全て未書き込み状態となったデータキャッシュ 1 3 b のブロックを解放した後、S 3 0 に戻る。

【0 1 6 6】

以上のように、図 6 (a) に示すフローチャートに従って処理を行うことにより、データキャッシュ 1 3 b 内の領域を、図 4 (e) に示す状態から図 4 (a)

に示す状態へと戻すことができ、一括して未使用になったデータキャッシュ 13 b 中の領域を解放できる。

【0167】

ここで、本実施形態の CPU 10 は、ライトバック制御を行っているため、キャッシュ 13 a ～ 13 c 中に記録された情報は、適宜外部メモリへと反映される。このため、キャッシュ 13 a ～ 13 c 中に記録された情報は、反映される前に消去されることがないように、一旦記録されるとメモリ領域を使用したままとする。なお、スタック領域に記録されたデータ等については、外部メモリへ反映させる必要はないことから、キャッシュ 13 a ～ 13 c 内の不要なデータ等が格納されたアドレスを検出して、対応するキャッシュ領域を解放することで、外部メモリに対する無駄なライトバックが行われることを防止できる。

【0168】

つぎに、タスクが終了した時の CPU 10 の処理について、図 6 (b) を用いて説明する。

【0169】

S 37 においては、タスク終了処理を行うと、処理が終了したタスクの不要なデータ等を格納したアドレスを検出し、対応するキャッシュ 13 a ～ 13 c 内の領域を解放するために、以下のような処理が進められる。

【0170】

すなわち、まず、S 38 において、データキャッシュ 13 b のスタック領域の全アドレスを、未書き込み状態に変更する。

【0171】

続いて、S 39 において、命令キャッシュ 13 a の使用領域の全ブロックを解放し、S 40 において、データキャッシュ 13 b のローカル変数の全アドレスを未書き込み状態に変更する。

【0172】

S 41 においては、データキャッシュ 13 b のローカル変数の全ブロックを解放する。

【0173】

2 以上の処理の後に、S 4 2において、Mフラグが「0」であるか否かが判定され、ここで、Mフラグが「0」である場合には処理を終了する。一方、Mフラグが「0」でない場合には、S 4 3に進む。

【0174】

S 4 3においては、残りのタスクで外部メモリが必要であるか否かが判定される。ここで、外部メモリが必要である場合には処理を終了し、外部メモリが必要でない場合にはS 4 4に進み、再度、図5（a）に示した外部メモリ停止モードへ移行する。

【0175】

以上のように、本実施形態のCPU 10は、外部メモリへのアクセス禁止モードへ移行することで、外部メモリに対する電力供給を停止したり、低消費電力モードに移行したりする等して、PC 1の消費電力を削減することができる。また、処理を継続したまま、外部メモリアクセス禁止モード、外部メモリアクセス許可モードの切り替えを行うことができる。さらに、外部メモリへのアクセスが必要な場合には、Mフラグを「1」として、外部メモリへのアクセス禁止を解除してライトバック制御を行い、キャッシュ13a～13c内のデータ等を外部メモリに書き込むことができるため、外部メモリに書き込まれたデータ等をキャッシュ13a～13cから消去して、キャッシュ13a～13cの残り容量を増加させ、キャッシュ13a～13cを効率よく利用できる。

【0176】

次に、データキャッシュ13bに記憶されたデータ等を、外部メモリに対してライトバックする動作について、図7（a）を用いて説明する。

【0177】

なお、ライトバックとは、キャッシュ13a～13cに蓄積された情報を所定のタイミングで外部メモリへと反映させる処理をいう。

【0178】

CPU 10の処理速度は、外部メモリへの書き込み速度よりも速いため、CPU 10が処理を行うたびに、外部メモリに反映させるべき情報が蓄積されていく。そこで、本実施形態のCPU 10においては、適宜タイミングを図って、以下

に説明する処理を行うことにより、データキャッシュ 13 b の情報を外部メモリへと反映させている。

【0179】

S 5 0 においては、M フラグが「0」であるか否かが判定される。M フラグが「0」でない場合には、外部メモリへのアクセス禁止が解除された状態であるため、S 5 6 に進む。一方、M フラグが「0」である場合には、S 5 1 に進み、キャッシュ 13 a ～ 13 c の空き容量が少ない状態であるか否かについて判定を行う。

【0180】

S 5 1 においてキャッシュ 13 a ～ 13 c の空き容量が少なくない場合には、CPU 1 0 は処理を終了する。一方、キャッシュ 13 a ～ 13 c の空き容量が少ない場合には、S 5 2 に進み、外部メモリが起動中であるか否かを判定する。

【0181】

ここで、外部メモリが起動中である場合には S 5 4 に進み、外部メモリが起動中でない場合には S 5 3 に進み、外部メモリを起動した後、S 5 4 に進む。

【0182】

S 5 4 においては、外部メモリの起動が完了したか否かに関する外部メモリ初期化完了の割り込みが入るまで待機する。ここで、割り込みが入った場合には、図 5 (c) を用いて説明したように、M フラグが「1」となる。そこで、S 5 5 においては、外部メモリの起動が完了したことを確認するために、M フラグが「0」であるか否かを判定して、M フラグが「0」でない場合にのみ S 5 6 に進み、M フラグが「0」の場合には S 5 5 を繰り返すようにしている。

【0183】

そして、外部メモリの起動完了を確認した後、S 5 6 において、データキャッシュ 13 b の内容を外部メモリに反映させる。

【0184】

S 5 7 においては、データキャッシュ 13 b において、外部メモリに反映させたデータ等が格納された領域のブロックを解放する。

【0185】

このようにデータキャッシュ 13 b から外部メモリへライトバックが行われ、データキャッシュ 13 b 内における対応する領域を他のデータのために利用すべくメモリ領域のブロックを解放することができる。

【0186】

そして、それまでデータキャッシュ 13 b 内で保持していたデータ等がパージされているため、外部メモリからデータキャッシュ 13 b 内に書き戻すことができるように、S 58 において、P フラグを「1」としてパージされたことを認識できるようにした後、処理を終了する。

【0187】

以上の動作によって、データキャッシュ 13 b の内容を外部メモリに反映させる、いわゆるライトバック制御を行うとともに、P フラグを「1」とすることで、データキャッシュ 13 b 内のデータ等がパージされたことを認識でき、必要な場合には、外部メモリへライトバックしたデータ等を書き戻すことができる。

【0188】

一方、P フラグが「0」である場合には、キャッシュ 13 a ～ 13 c 内のデータ等はパージされていないデータ等が検出されたことを意味する。よって、P フラグが「0」である場合には、キャッシュ 13 a ～ 13 c 内に存在する同じデータ等の外部メモリからの書き戻しを防止して、無駄な処理が行われることを防止できる。

【0189】

次に、キャッシュミスの場合の動作について、図 7 (b) を用いて説明する。

【0190】

なお、キャッシュミスとは、初めからキャッシュ 13 a ～ 13 b に格納されていないと分かっている外部メモリにデータを読みに行く図 5 (b) の S 10 のようなステップとは異なり、必要なデータをキャッシュ 13 a ～ 13 b に読みに行ったが、キャッシュ 13 a ～ 13 c 内に存在しなかった場合をいう。

【0191】

先ず、S 59 において、M フラグが「0」であるか否かを判定する。

【0192】

ここで、Mフラグが「0」でない場合には、外部メモリがアクセス可能な状態であるため、S64に進む。

【0193】

一方、Mフラグが「0」である場合には、S60において、外部メモリが起動中であるか否かについて判定する。外部メモリが起動中の場合には、S62に進む。ここで、外部メモリが起動中でない場合には、S61において外部メモリを起動する。

【0194】

S62においては、外部メモリを起動した後、外部メモリ初期化完了の割り込みが入るまで待機する。ここで、割り込みが入った場合には、外部メモリへのアクセス禁止が解除され、Mフラグが「1」となる。そこで、S63においては、Mフラグが「0」であるか否かを判定して、Mフラグが「0」でない場合にのみS64に進み、Mフラグが「0」の場合にはS63を繰り返すようにしている。これにより、外部メモリの起動が完了するまで待機することができる。

【0195】

S64では、データキャッシュ13bの残り容量が少ない状態であるか否かについて判定を行う。キャッシュ13bの残り容量が充分ある場合にはS69に進む。

【0196】

一方、データキャッシュ13bの残り容量が少ない場合には、S65に進み、ライトバックが必要であるか否かについて判定を行う。

【0197】

ここで、ライトバックが不要である場合にはS67に進み、一方、ライトバックを行ってデータキャッシュ13bの空き容量を確保する必要がある場合には、S66において、データキャッシュ13bから外部メモリへのライトバックを行う。

【0198】

S67においては、例えば、ライトバックやその他の処理によって、不要となったデータ等に対応するデータキャッシュ13b内のブロックを解放する。

【0199】

S68においては、Pフラグを「1」とし、データキャッシュ13b内のデータ等がパージされたことを認識できるようにする。

【0200】

S69においては、外部メモリから所望のデータ等をデータキャッシュ13bへ取り込んだ後、処理を終了する。

【0201】

以上のように、本実施形態のCPU10は、予めキャッシュにロードしたプログラムと、キャッシュで利用可能なアドレス空間に割り付けた変数とを用いて、できる限り外部メモリにアクセスすることなく、内部のキャッシュ13a～13cだけを用いて処理を進める。そして、キャッシュ13a～13cだけでは処理しきれずに、外部メモリが必要になった時点で、それまでの処理を継続しつつ、外部メモリを起動している。

【0202】

これにより、外部メモリに対するアクセス禁止期間をできる限り長くして、省電力化を図るとともに、外部メモリを用いずにキャッシュ13a～13cだけで処理を進めることで、高速処理を実現したCPU10およびそれを備えたPC1を提供できる。

【0203】

本発明のCPUの制御方法は、以上のように、図5(a)～図7(b)に示すフローチャートに従ってCPU10を制御することにより、できる限りキャッシュ13a～13cのみを用いて処理を行うとともに、外部メモリへのアクセスを禁止することで、外部メモリへの電力供給を停止する、あるいは低消費電力モードへ移行する等して、PC1の電力消費量を削減できる。

【0204】

なお、本実施形態のCPU10では、キャッシュ13a～13cの記憶容量が満杯であるか否かについて判定し、外部メモリへのアクセス禁止モードへ移行するか否かを決定する例を挙げて説明したが、本発明はこれに限定されるものではない。例えば、キャッシュ13a～13cにおける空き容量を検出し、これと処

理を行うタスクの必要メモリ量とを比較して、外部メモリへのアクセスが必要であるか否かを決定してもよい。

【0205】

ただし、本実施形態のようにキャッシュ13a～13cの空き容量だけを検出して判断することで、空き容量が満杯である場合にはタスクの必要メモリ量とは無関係に外部メモリへのアクセスが必要となるため、外部メモリへのアクセスの必要性の有無を認識できる。

【0206】

また、本実施形態のCPU10では、外部メモリへのアクセス禁止状態を「Mフラグ」を用いて設定し、キャッシュ13a～13c内のデータ等がパージされたか否かを「Pフラグ」を用いて認識していたが、このようなフラグの使用は本発明を実施するための一つの手段として例示したものであり、本発明はこれに限定されるものではない。

【0207】

また、本実施形態のPC1は、上述したように、外部ROM30、外部RAM40とともに、電源20を備えている。これにより、キャッシュ13a～13cだけを用いてタスクの処理が可能な場合、すなわち、外部メモリへのアクセスが禁止されている場合には、電源20から外部ROM30および外部RAM40に対する電源供給を停止して、PC1の消費電力を削減できる。

【0208】

また、本実施形態のPC1は、上述したように、外部メモリとして、外部ROM30と外部RAM40とを備えており、両方の記憶装置の特性に応じて両者を使い分けている。これにより、本実施形態のCPU10によって処理を行うことにより、より効果的にPC1の消費電力を低減することができる。

【0209】

なお、本発明のCPUは、2次キャッシュ、3次キャッシュ等の設置により将来的にキャッシュの容量が増大してきた場合には、低消費電力化、高速処理を実現するCPUとして特に有効である。

【0210】

【発明の効果】

本発明のCPUは、以上のように、処理するタスクの必要メモリ量に応じて、上記キャッシュだけにアクセスしてタスクの処理が可能であるか否かを判定し、該処理が可能である場合には、上記外部メモリに対するアクセスを禁止する制御手段を備えている構成である。

【0 2 1 1】

それゆえ、CPU内に備えられているキャッシュだけを用いてタスクの処理を行うことができる状態においては、外部メモリへのアクセスを禁止しているため、外部メモリに対する電力供給を停止する、あるいは低消費電力モードに切り替える等の措置を取ることができ、結果として、本発明のCPUを備えた情報処理装置の消費電力を削減することができるという効果を奏する。

【0 2 1 2】

さらに、キャッシュよりも低速な外部メモリに対するアクセス回数を従来よりも減らし、高速アクセスが可能なキャッシュを用いて処理を行うため、迅速な処理が可能になる。

【0 2 1 3】

上記制御手段は、上記キャッシュの残り容量および／または処理するタスクの必要メモリ量を検出することがより好ましい。

【0 2 1 4】

それゆえ、例えば、キャッシュの残り容量とタスクの必要メモリ量とを検出し、両者を比較することで、キャッシュのみを用いて該タスクの処理を行うことが可能であるか否かを判定できるという効果を奏する。

【0 2 1 5】

上記外部メモリに対するアクセスが禁止されている状態において、上記制御手段が、上記キャッシュだけでは処理が不可能と判断した場合、あるいはキャッシュミスが発生した場合には、上記外部メモリに対するアクセス禁止を解除することがより好ましい。

【0 2 1 6】

それゆえ、外部メモリに対するアクセス禁止状態において、アクセス禁止を解

除することで、ライトバック制御を行って、キャッシュ内のデータ等を外部メモリに書き込むことができるため、外部メモリに書き込まれたデータ等をキャッシュから消去して、キャッシュの残り容量を増加させることができるという効果を奏する。

【0 2 1 7】

内部クロックのクロック周波数を制御するクロック制御手段を備えており、該クロック制御手段は、上記外部メモリに対するアクセスが禁止された場合には、上記クロック周波数を変化させることがより好ましい。

【0 2 1 8】

それゆえ、例えば、外部メモリへのアクセスが禁止されている場合、すなわち、高速アクセスが可能なキャッシュのみにアクセスして処理を行っている場合には、クロック周波数を低くすることにより、CPUの消費電力を削減することができるという効果を奏する。

【0 2 1 9】

上記制御手段は、上記キャッシュ内における不要なデータを格納しているアドレスを検出し、該検出されたアドレスに対応するキャッシュ領域を解放することがより好ましい。

【0 2 2 0】

それゆえ、例えば、処理の終了したタスク専用のプログラム、データの領域、スタックの不要なデータ等を格納したアドレスを検出し、このデータ等に対応するキャッシュ領域を、データが書き込まれていない領域とみなして解放することにより、実質的にキャッシュの空き容量を増やすことができるため、キャッシュを有効に利用できるという効果を奏する。

【0 2 2 1】

電源投入後の初期動作時においては、所望のプログラムおよびデータを上記外部メモリから上記キャッシュに読み込んだ上で、上記外部メモリへのアクセスを禁止することがより好ましい。

【0 2 2 2】

それゆえ、CPUが備えているキャッシュのみにアクセスして処理を行うこと

ができるとともに、外部メモリへのアクセスを禁止して外部メモリに対する電力供給を停止することで、初期動作時から消費電力を削減した状態にできるという効果を奏する。

【0223】

上記制御手段は、タスクの状況が変化する際に、上記外部メモリに対するアクセスが必要か否かについての判断を行うことがより好ましい。

【0224】

それゆえ、OS等によって管理されているタスクの状況が変化した場合には、外部メモリへのアクセスが必要になる可能性が生じるため、その時点で外部メモリを使用する必要があるか否かについて判断し、外部メモリが必要な場合には、実際にアクセスが発生する前に外部メモリを起動することができるため、外部メモリの起動待ち時間を短縮して、処理を迅速に行うことができるという効果を奏する。

【0225】

上記制御手段は、上記キャッシュ内のプログラムおよびデータについて、パージされたか否かについて判定し、パージされていない場合には、上記プログラムおよびデータを上記外部メモリから読み込む処理を行わないことがより好ましい。

【0226】

それゆえ、キャッシュ内に書き込まれたプログラムやデータについて、これらがパージされていない場合には、該プログラム等はキャッシュ内に残っているため、キャッシュのみで処理を行うモードに移行する場合に必要なプログラム等を、新たに外部メモリから読み込む処理を省略して、無駄な処理を減らすことができるという効果を奏する。

【0227】

本発明の情報処理装置は、以上のように、上記電源供給手段は、上記外部メモリへのアクセスが禁止されている場合には、該外部メモリへの電力供給を停止する構成である。

【0228】

それゆえ、外部メモリへのアクセスが禁止されている場合、すなわちCPUが内部に備えているキャッシュに対してアクセスするだけで処理を継続できる場合には、外部メモリへの電力供給を停止することで、情報処理装置の消費電力を削減できるという効果を奏する。

【0 2 2 9】

上記外部メモリは、複数のモジュールを含んでおり、上記制御手段は、上記モジュールごとに電力供給の制御を行うことがより好ましい。

【0 2 3 0】

それゆえ、外部メモリが、例えば、RAM、ROM等の複数のモジュールからなる場合には、このモジュールの性質に応じて電力供給を制御することで、さらに効率的に情報処理装置の消費電力を低減できるという効果を奏する。

【0 2 3 1】

本発明のCPUの制御方法は、以上のように、処理するタスクの必要メモリ量に応じて、上記キャッシュだけにアクセスしてタスクの処理が可能であるか否かを判定し、該処理が可能であると判定した場合には、上記外部メモリに対するアクセスを禁止する制御方法である。

【0 2 3 2】

それゆえ、CPU内に備えられているキャッシュだけを用いてタスクの処理を行うことができる状態においては、外部メモリへのアクセスを禁止しているため、外部メモリに対する電力供給を停止させて消費電力を削減できるという効果を奏する。

【0 2 3 3】

さらに、キャッシュよりも低速な外部メモリに対するアクセス回数を従来よりも減らし、高速アクセスが可能なキャッシュを用いて処理を行うため、迅速な処理が可能になる。

【図面の簡単な説明】

【図 1】

本発明の一実施形態にかかるCPUの内部構造を示すブロック図である。

【図 2】

図1のCPUと外部メモリとを備えたPCの構成を示すブロック図である。

【図3】

(a)は、命令キャッシュ、データキャッシュにおける使用状況（既使用容量）の時間変化を模式的に示すタイミングチャートであり、(b)は、(a)に示すタイミングチャートに続く状態の一例を示すタイミングチャートである。

【図4】

(a)～(e)は、タスク開始前またはタスク終了後におけるスタックの使用状態を示す図であり、(f)は、この場合の実際のプログラムを示す図である。

【図5】

(a)～(e)は、図1のCPUによる処理を示すフローチャートである。

【図6】

(a)・(b)は、データキャッシュ解放、タスク終了時のCPUの処理を示すフローチャートである。

【図7】

(a)・(b)は、データキャッシュライトバック、キャッシュミス発生時のCPUの処理を示すフローチャートである。

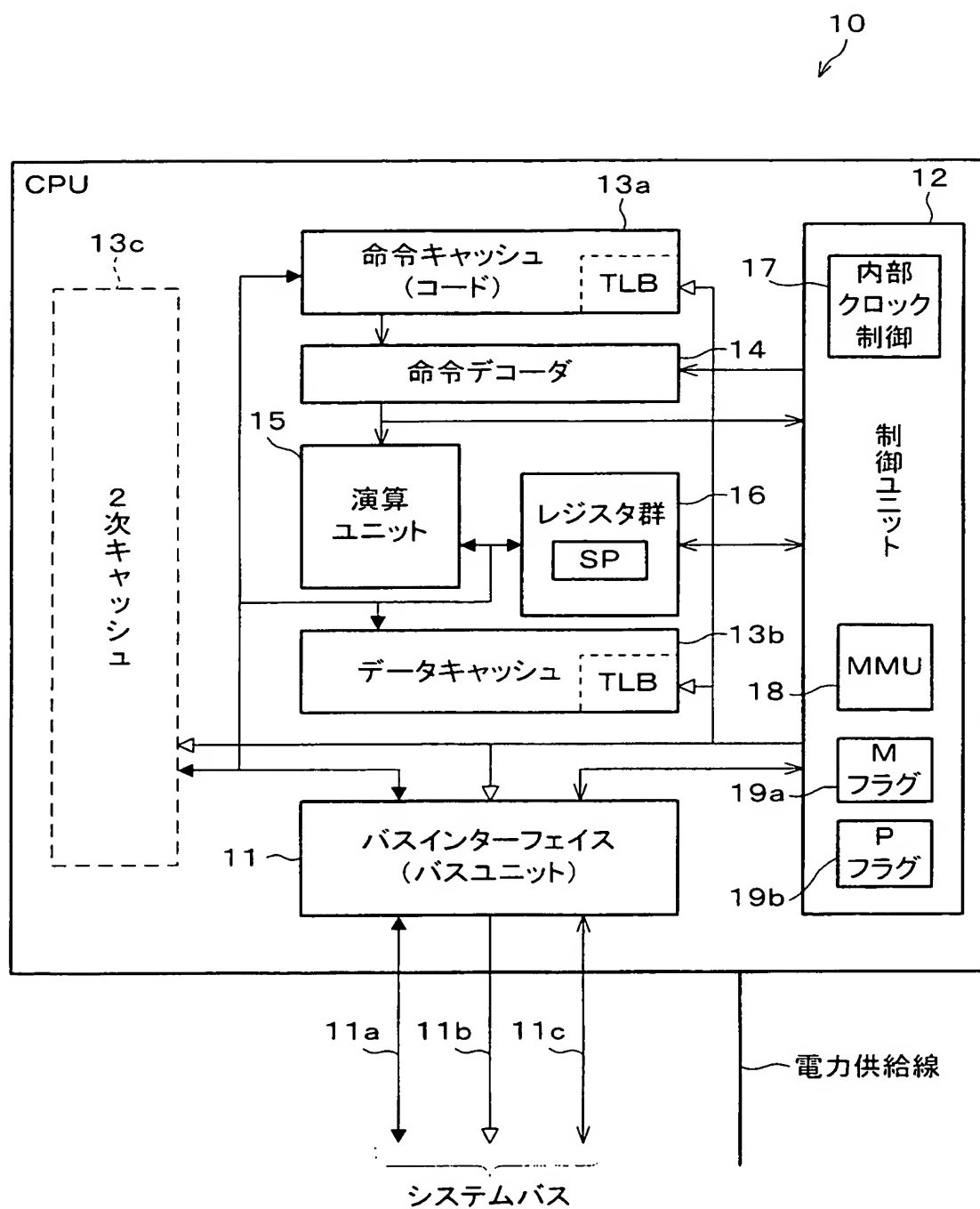
【符号の説明】

- 1 PC（情報処理装置）
- 10 CPU
- 11 バスインターフェース
- 12 制御ユニット（制御手段）
- 13a 命令キャッシュ（キャッシュ）
- 13b データキャッシュ（キャッシュ）
- 13c 2次キャッシュ（キャッシュ）
- 14 命令デコーダ
- 15 演算ユニット
- 16 レジスタ群
- 17 内部クロック制御部（内部クロック制御手段）
- 18 MMU

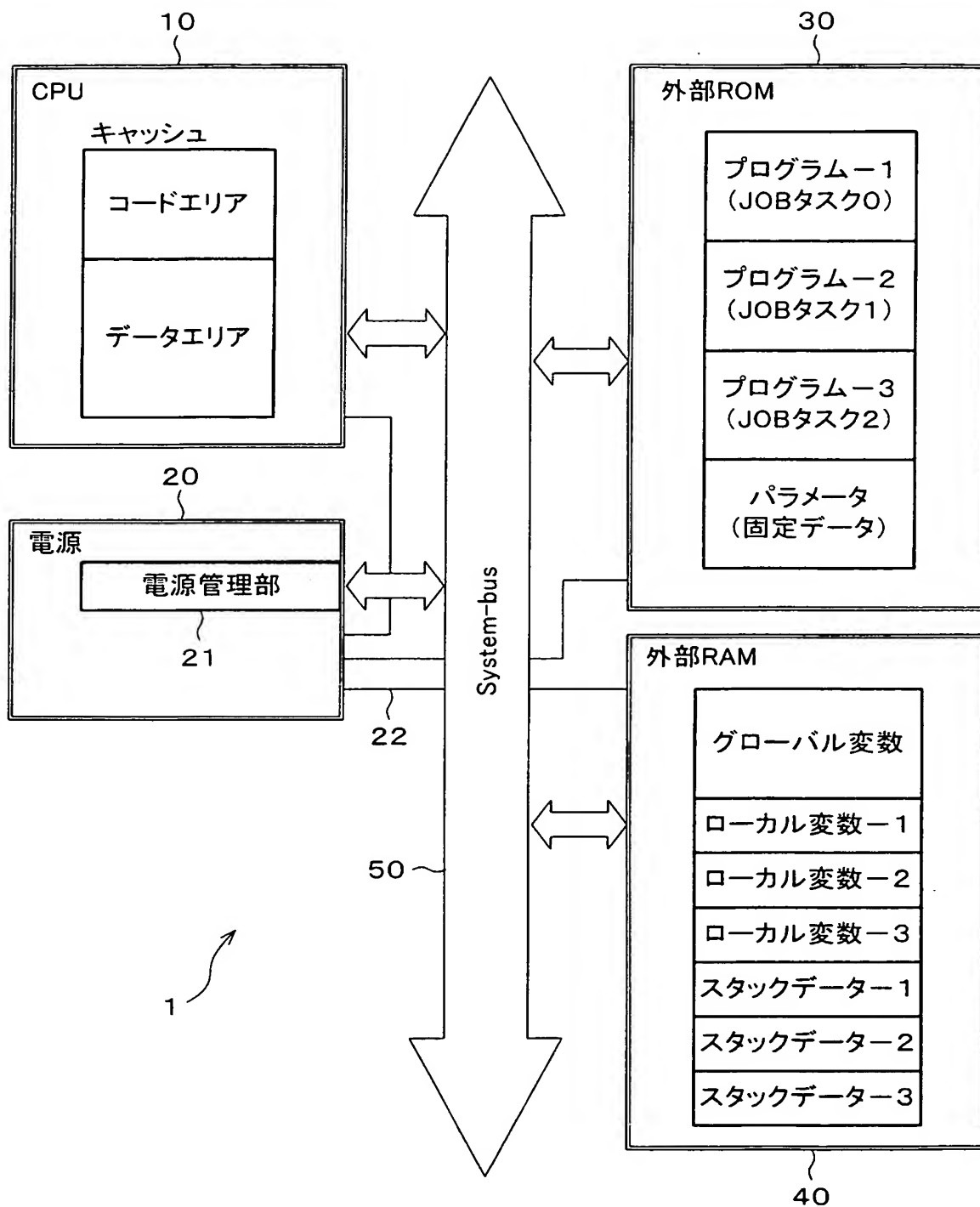
- 1 9 a Mフラグ
- 1 9 b Pフラグ
- 2 0 電源（電源供給手段）
- 2 1 電源管理部
- 2 2 電力供給線
- 3 0 外部R O M（外部メモリ）
- 4 0 外部R A M（外部メモリ）
- 5 0 システムバス

【書類名】 図面

【図 1】

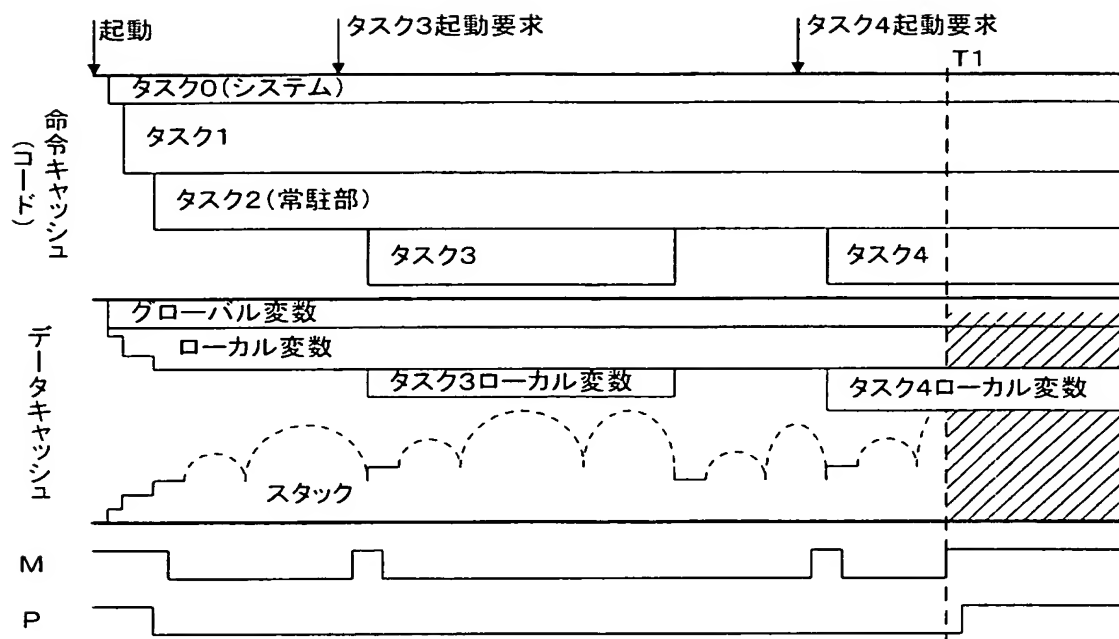


【図 2】

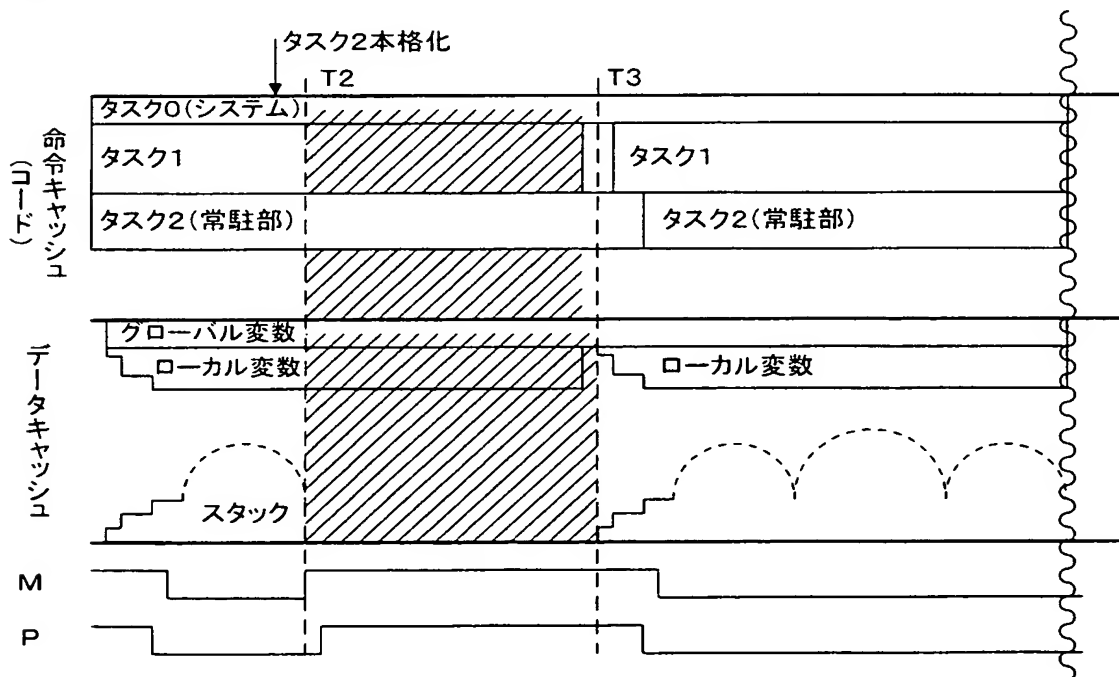


【図 3】

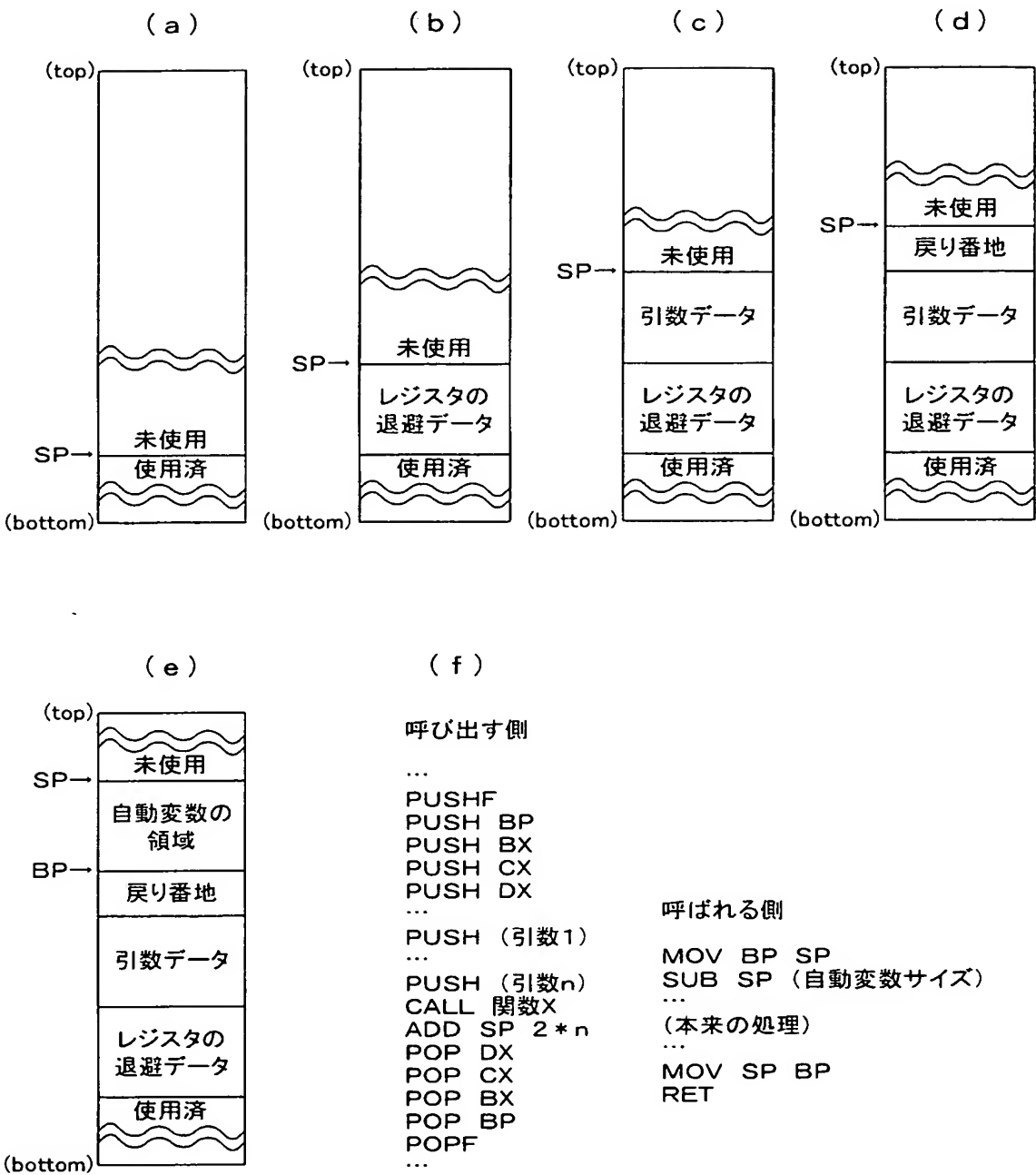
(a)



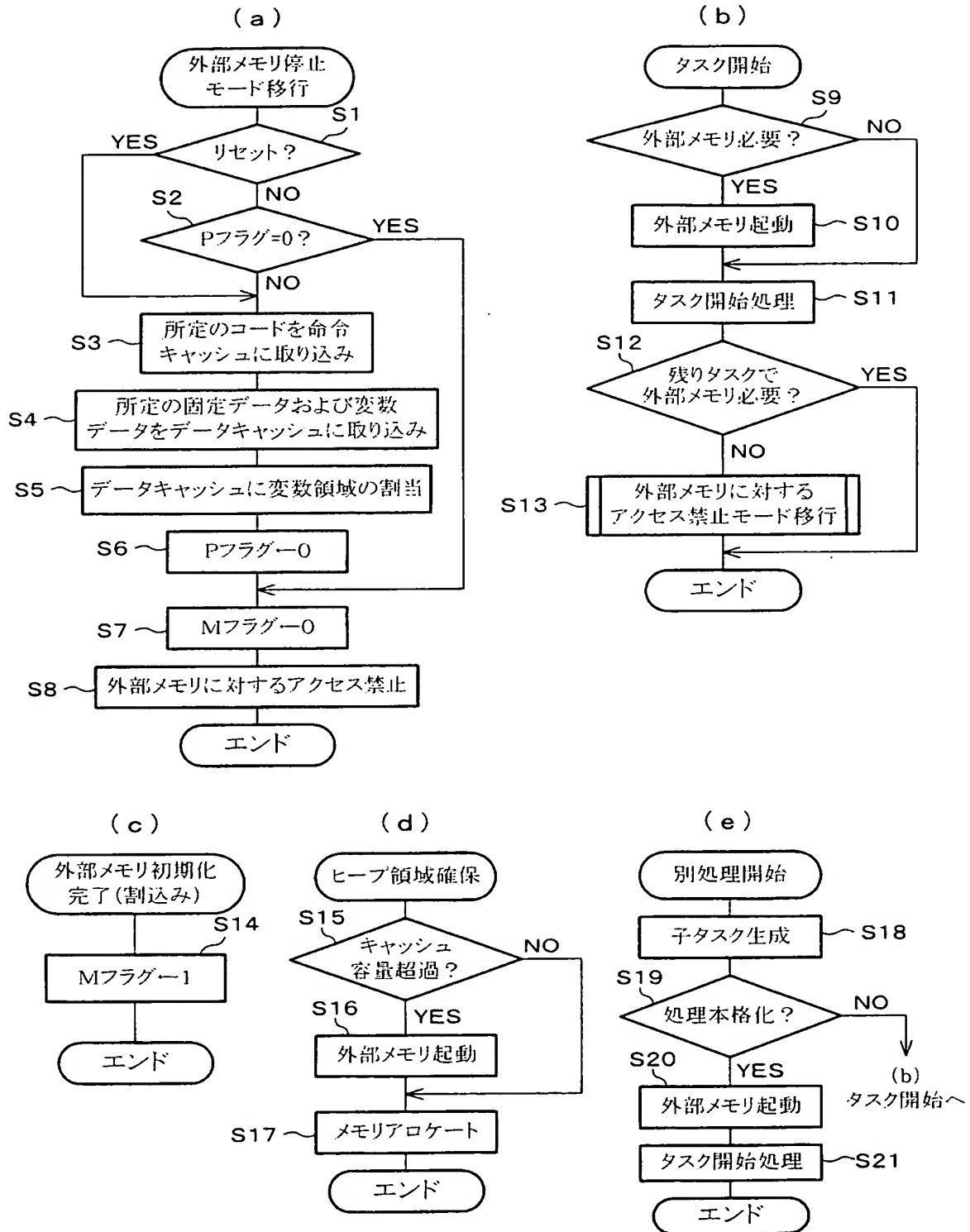
(b)



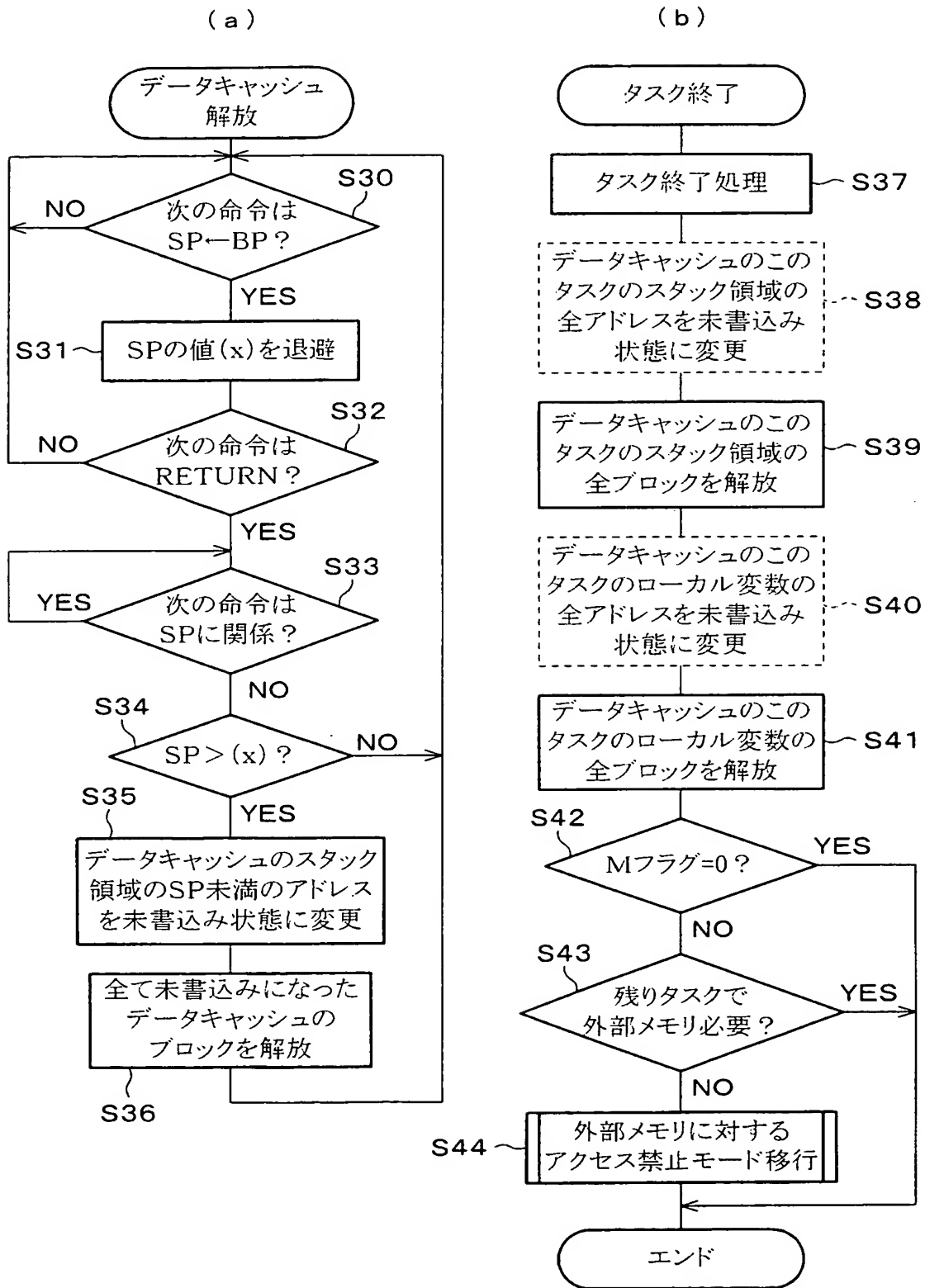
【図 4】



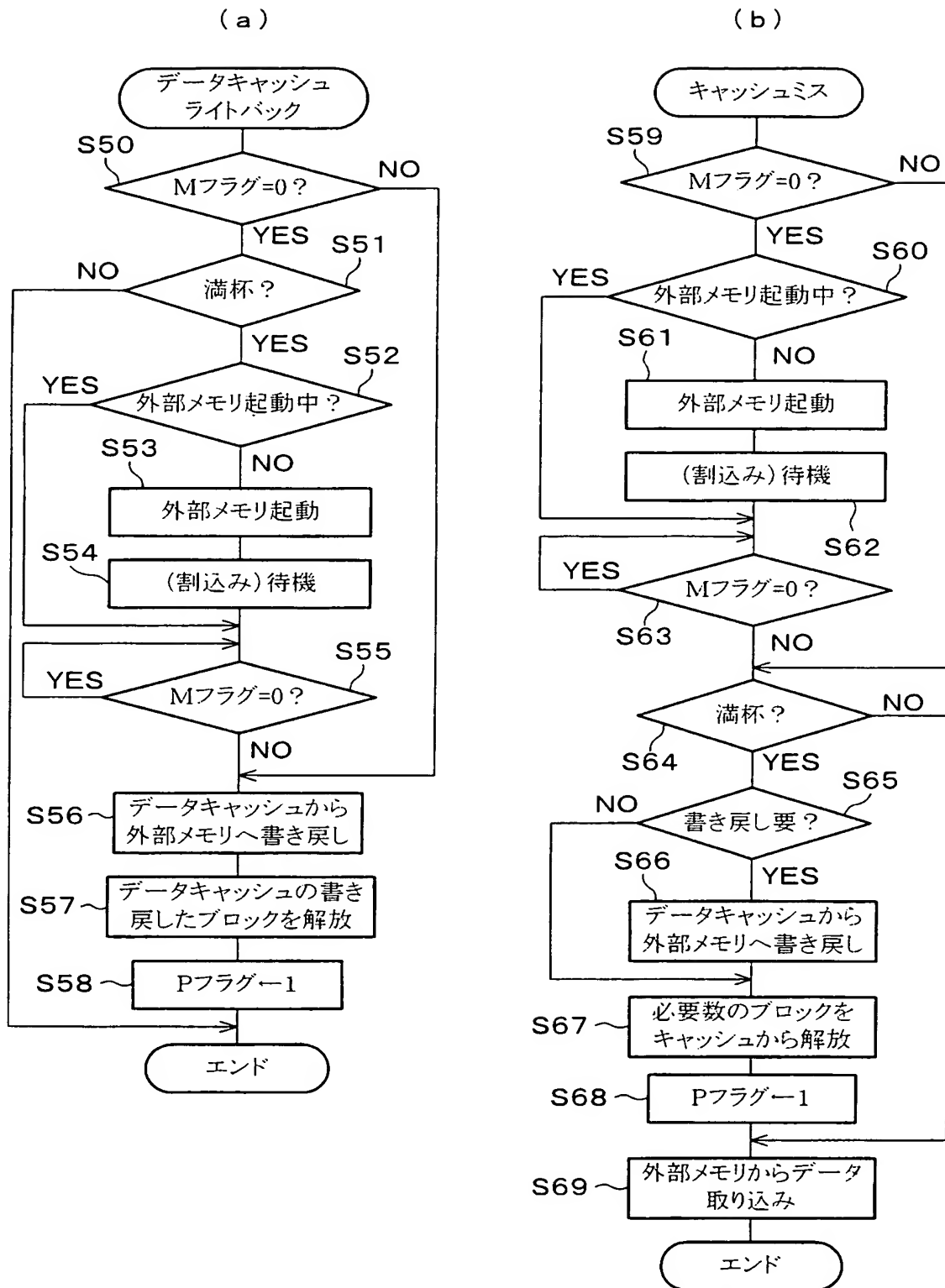
【図5】



【図 6】



【図 7】



【書類名】 要約書

【要約】

【課題】 外部メモリを使用せずにCPU内部で処理を行うことで、迅速な処理が可能になるとともに、外部メモリに対する電力供給を停止し、消費電力を低減することができるCPUおよびこれを備えた情報処理装置、CPUの制御方法を提供する。

【解決手段】 CPU10は、バスインターフェース11、制御ユニット12、命令キャッシュ13a、データキャッシュ13b、2次キャッシュ13c、命令デコーダ14、演算ユニット15およびレジスタ群16を備えている。CPU10は、キャッシュ13a～13cのみで処理が可能である場合には、外部メモリに対するアクセスを禁止し、外部メモリへの電力供給を停止する。

【選択図】 図1

特願 2 0 0 2 - 3 4 0 0 5 5

出 願 人 履 歴 情 報

識別番号

[0 0 0 0 0 5 0 4 9]

1 . 変更年月日

1 9 9 0 年 8 月 2 9 日

[変更理由]

新規登録

住 所

大阪府大阪市阿倍野区長池町 2 2 番 2 2 号

氏 名

シャープ株式会社